

Processadors de Llenguatge

**ESTUDI DELS REQUISITS
D'UN ALGORISME D'ANÀLISI
SINTÀCTICA DESCENDENT**

Juan Antonio Pérez Ortiz

Copyright (c) 2003 Juan Antonio Pérez Ortiz

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, version 1.2, or any later version published by the Free Software Foundation; with no Invariant Section, no Front-Cover Texts, and no Back-Cover Texts.

Es concedeix permís per a copiar, distribuir i/o modificar aquest document d'acord amb les condicions de la Llicència de Documentació Lliure de GNU, versió 1.2, o qualsevol versió posterior publicada per la Free Software Foundation; sense cap secció invariant, cap text de portada ni cap text de contraportada.

De l'arrel a les fulles

Podem escriure un algorisme que construisca un arbre d'anàlisi sintàctica de l'arrel a les fulles llegint l'entrada d'esquerra a dreta?

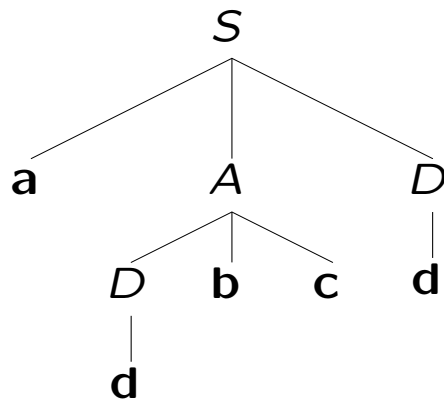
Quins en serien els requeriments?

Siga G_0 :

$$\begin{array}{lcl} S & \longrightarrow & \mathbf{a A D} \\ A & \longrightarrow & \mathbf{D b c} \mid \mathbf{e} \\ D & \longrightarrow & \mathbf{d} \end{array}$$

Cadena d'entrada: **a d b c d \$**

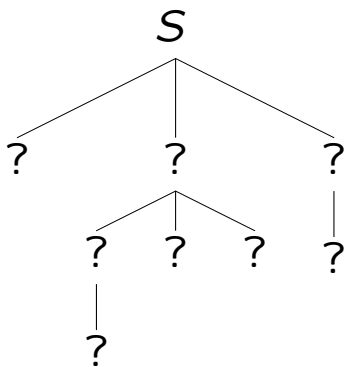
Objectiu:



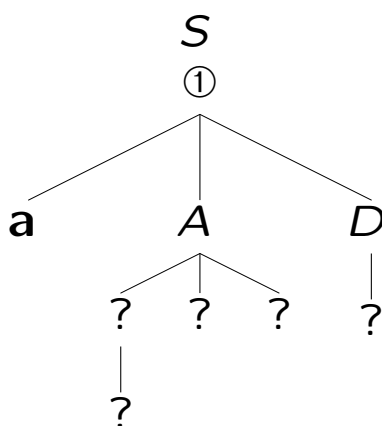
- Dues accions bàsiques:
 - Llegir un terminal (un *token*)
 - Descendir pel arbre

Traça de l'anàlisi descendent

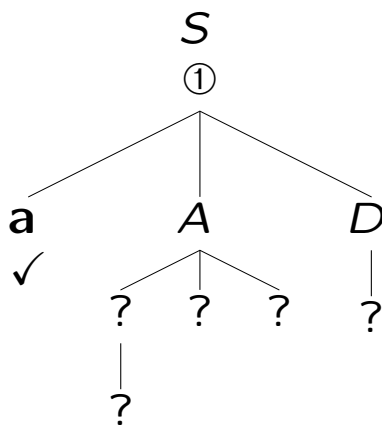
Pas 1. Començar per l'arrel:



Pas 2. Descendir per S ①:

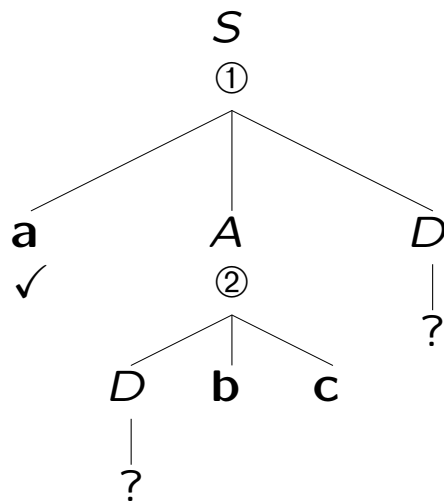


Pas 3. Llegir **a** :

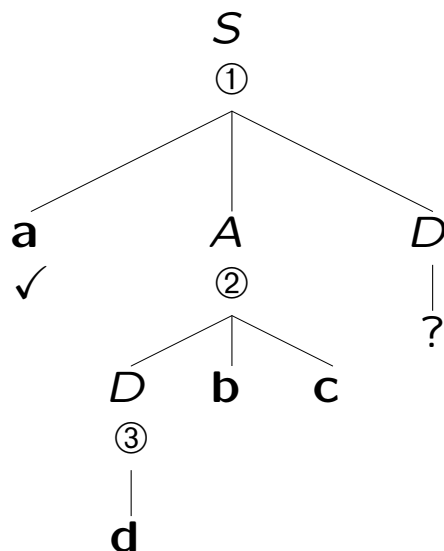


Traça de l'anàlisi descendent

Pas 4. Descendir per A ② (prenent una decisió!):

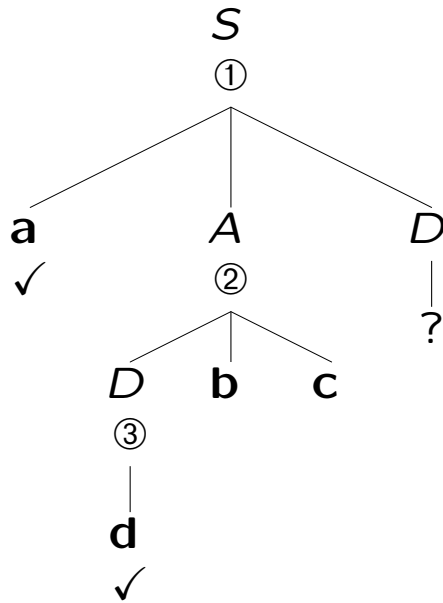


Pas 5. Descendir per D ③:

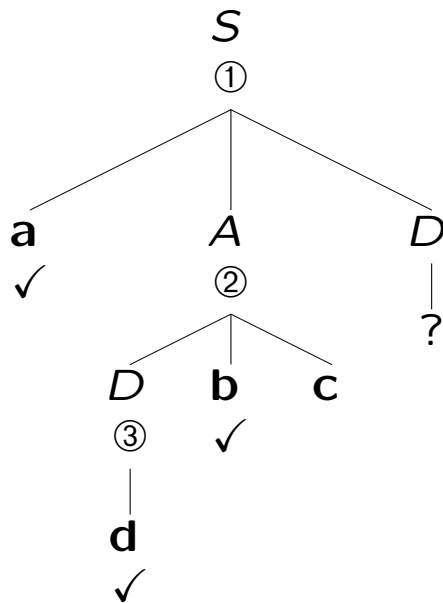


Traça de l'anàlisi descendent

Pas 6. Llegir **d** :

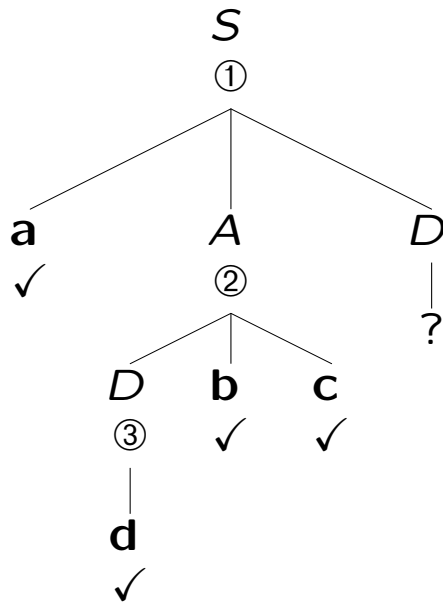


Pas 7. Llegir **b** :

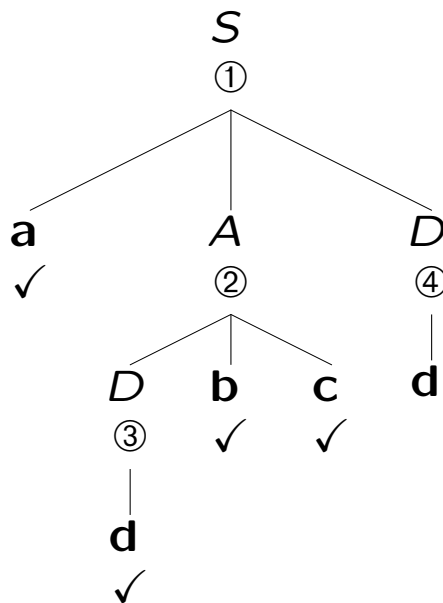


Traça de l'anàlisi descendent

Pas 8. Llegir c :

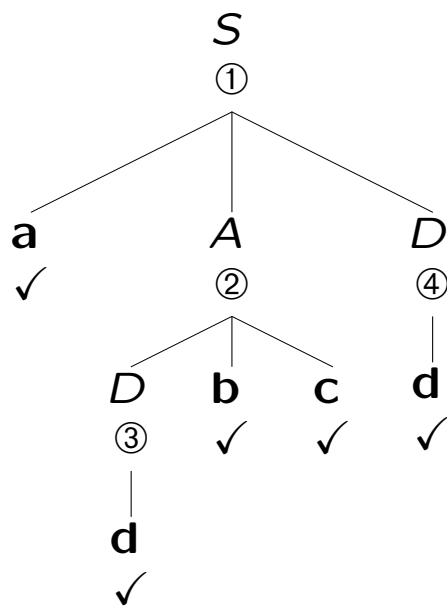


Pas 9. Descendir per D ④:



Traça de l'anàlisi descendent

Pas 10. Llegir **d** :



Esbós núm. 0

- Dues accions bàsiques:
 - Llegir \equiv aparellar
 - Descendir \equiv expandir

- Si el primer node pendent a l'arbre és un terminal, intentem llegir-lo (aparellar-lo) amb el següent símbol de l'entrada.

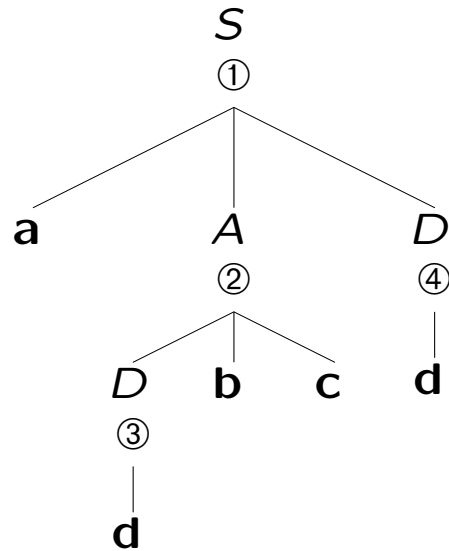
- Si és un no terminal, hi descendim (expandim) immediatament; tot i que encara no sabem quina decisió prendre quan hi ha més d'una possibilitat d'expansió...

- No podem "oblidar" cap germà pendent. Cal tenir un lloc on anar emmagatzemant-los; una pila és adequada.

- L'algorisme fa un recorregut per l'esquerra de l'arbre d'anàlisi sintàctica.

La pila d'un analitzador descendent

Àrbre d'anàlisi sintàctica:



Traça:

Pila	Entrada	Acció
\$ S	a d b c d \$	expandir $S \rightarrow a A D$ ①
\$ D A a	a d b c d \$	emparellar
\$ D A	d b c d \$	expandir $A \rightarrow D b c$ ②
\$ D c b D	d b c d \$	expandir $D \rightarrow d$ ③
\$ D c b d	d b c d \$	emparellar
\$ D c b	b c d \$	emparellar
\$ D c	c d \$	emparellar
\$ D	d \$	expandir $D \rightarrow d$ ④
\$ d	d \$	emparellar
\$	\$	acceptar

En ② podem expandir per $A \rightarrow D b c$ o per $A \rightarrow e$. Com prenem la decisió correcta, si no volem haver de tornar enrere?

Accions de l'anàlisi descendent predictiva

- Emparellar **a** : si el símbol de preanàlisi és **a** , desapilar **a** i actualitzar el preanàlisi al següent símbol de l'entrada.
- Expandir per $A \rightarrow \alpha$: desapilar A i apilar els símbols de α en ordre invers. En el cas de l'anàlisi sintàctica predictiva, només s'ha de poder considerar una única regla d'expansió en un context determinat.

Esbós núm. 1

- En començar, apilar \$.
- Siga q l'element del cim de la pila.
 - Si q es un terminal, intentar emparellar-lo amb el símbol de preanàlisi.
 - Si q es un no terminal A , expandir per $A \rightarrow \alpha$. Pot haver diferents opcions i triar-ne l'adequada no sempre és trivial.
- Acceptar si q és \$ i el preanàlisi és \$.

La taula d'anàlisi sintàctica descendent

- La qüestió més important es com decidir quina regla aplicar en una expansió quan n'hi ha més d'una possible.
- Necessitem una taula que ens done una informació similar a la següent (més endavant, aprendrem a construir-la):

$$T [S , \mathbf{a}] = \{ S \rightarrow \mathbf{a} A D \}$$

$$T [A , \mathbf{d}] = \{ A \rightarrow D \mathbf{b} \mathbf{c} \}$$

$$T [A , \mathbf{e}] = \{ A \rightarrow \mathbf{e} \}$$

$$T [D , \mathbf{d}] = \{ D \rightarrow \mathbf{d} \}$$

La resta d'entrades indiquen una situació d'error.

- D'això se'n diu taula d'anàlisi sintàctica i s'indexa mitjançant el no terminal del cim de la pila i el símbol de preanàlisi.
- Estudiarem, doncs, el cas que el preanàlisi només incloga el següent símbol de l'entrada, però no pas el segon, ni el tercer, etc.
- Per a qualsevol no terminal A i qualsevol terminal \mathbf{a} , ens interessa que $T[A, \mathbf{a}]$ continga una regla o cap. En altre cas, hauríem de considerar un algorisme amb marxa enrere.

No terminals amb més d'una regla

Encara no hem solucionat quina regla triem quan el no terminal del cim de la pila en té més d'una associada. Ara ens dedicarem a esbrinar-ho.

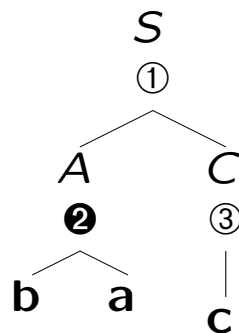
G_1 :

$$\begin{array}{l} S \longrightarrow A C \\ A \longrightarrow a C \mid b a \\ C \longrightarrow c \end{array}$$

G_1 no es ambigua.

Cadena: **b a c \$**

Arbre d'anàlisi sintàctica:



Traça:

Pila	Entrada	Acció
\$ S	b a c \$	expandir $S \rightarrow A C$
\$ C A	b a c \$	expandir $A \rightarrow a C$ o $A \rightarrow b a$?
...

No terminals amb més d'una regla

G_1 :

$$\begin{aligned} S &\longrightarrow AC \\ A &\longrightarrow aC \mid ba \\ C &\longrightarrow c \end{aligned}$$

Traça quan s'expandeix $A \rightarrow aC$:

Pila	Entrada	Acció
\$ S	b a c \$	expandir $S \rightarrow AC$
\$ C A	b a c \$	expandir $A \rightarrow aC$
\$ C C a \$	b a c \$	no podem seguir!

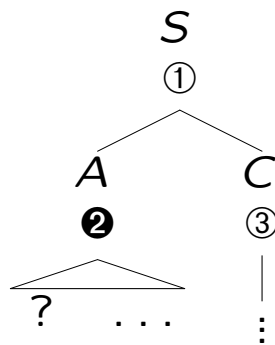
Traça quan s'expandeix $A \rightarrow ba$:

Pila	Entrada	Acció
\$ S	b a c \$	expandir $S \rightarrow AC$
\$ C A	b a c \$	expandir $A \rightarrow ba$
\$ C a b	b a c \$	emparellar
\$ C a	a c \$	emparellar
\$ C	c \$	expandir $C \rightarrow c$
\$ c	c \$	emparellar
\$	\$	acceptar

Com podem assegurar-nos que prenem la decisió correcta, si a més podem consultar un símbol de preanàlisi?

No terminals amb més d'una regla

No té sentit triar la regla $A \rightarrow \mathbf{a} C$ perquè ens trobaríem amb \mathbf{a} en el cim de la pila i \mathbf{b} en la entrada i no podríem emparellar-los.



Si el preanàlisi és \mathbf{b} , hem de triar l'opció (si n'hi ha) que deixi precisament \mathbf{b} al lloc del signe d'interrogació.

Doncs, ens quedem amb $A \rightarrow \mathbf{b} a$, perquè comença per \mathbf{b} .

Més enllà de la regla

De vegades, no n'hi ha prou amb mirar els prefixos de les parts dretes de les regles candidates i cal anar més enllà.

G_2 :

$$\begin{array}{lcl}
 S & \longrightarrow & A D \\
 A & \longrightarrow & C a \mid b a \\
 C & \longrightarrow & D c \\
 D & \longrightarrow & d
 \end{array}$$

Cadena: **d c a d \$**

Traça:

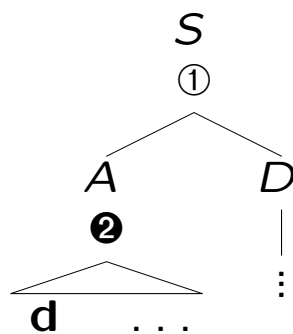
Pila	Entrada	Acció
\$ S	d c a d \$	expandir $S \rightarrow A D$
\$ D A	d c a d \$	expandir $A \rightarrow C a$ o $A \rightarrow b a$?
...

Més enllà de la regla

G_2 :

$$\begin{array}{lcl} S & \longrightarrow & A D \\ A & \longrightarrow & C a \mid \mathbf{b a} \\ C & \longrightarrow & D c \\ D & \longrightarrow & \mathbf{d} \end{array}$$

Com que el preanàlisi és \mathbf{d} , seguint el plantejament anterior, ens interessa la regla que done un arbre com el següent:



Això només és possible amb $A \rightarrow C a$, però hem de seguir la pista durant un parell de regles més per adonar-nos-en.

Conjunts de predicció

G_3 :

$$\begin{array}{lcl}
 S & \longrightarrow & A C \\
 A & \longrightarrow & C a \mid b a \\
 C & \longrightarrow & D c \\
 D & \longrightarrow & d \mid e D
 \end{array}$$

Considerem la següent situació d'anàlisi:

Pila	Entrada	Acció
...
\$... A	<div style="border: 1px solid black; display: inline-block; padding: 2px 5px;">?</div> ... \$...
...

Si tenim A en el cim de la pila i d o e com a preanàlisi, expandirem $A \rightarrow C a$; si el preanàlisi és b , aleshores expandirem $A \rightarrow b a$; en altre cas, emetrem un error.

Aixó ho escriurem com a conjunts de predicció així:

$$\text{PRED}(A \rightarrow C a) = \{ d, e \}$$

$$\text{PRED}(A \rightarrow b a) = \{ b \}$$

Conjunts de predicció

En l'exemple anterior:

$$\text{PRED} (A \rightarrow C \mathbf{a}) = \{ \mathbf{d}, \mathbf{e} \}$$

$$\text{PRED} (A \rightarrow \mathbf{b} \mathbf{a}) = \{ \mathbf{b} \}$$

Els conjunts de predicció són la base per a construir la taula d'anàlisi descendent:

$$T[A, \mathbf{a}] = \text{error!}$$

$$T[A, \mathbf{b}] = \{ A \rightarrow \mathbf{b} \mathbf{a} \}$$

$$T[A, \mathbf{c}] = \text{error!}$$

$$T[A, \mathbf{d}] = \{ A \rightarrow C \mathbf{a} \}$$

$$T[A, \mathbf{e}] = \{ A \rightarrow C \mathbf{a} \}$$

$$T[A, \$] = \text{error!}$$

...

Recordem que per a poder fer l'anàlisi predictiva, cada cel·la de T ha de contenir només una regla o cap. Dit d'una altra manera, la intersecció dels conjunts de predicció de les regles amb la mateixa part esquerra ha de ser buida.

Exercici: calcular T per a la resta de no terminals de G_3 .

Esbós núm. 2

- En començar, apilar \$.
- Siga q l'element del cim de la pila.
 - Si q es un terminal, intentar emparellar-lo amb el símbol de preanàlisi.
 - Si q és un no terminal A i el preanàlisi és a , expandir per la (única) regla $A \rightarrow \alpha$ que inclou a en el conjunt de predicció, és a dir, per $T[A, a]$; si no n'existeix, cal emitir un error.
- Acceptar si q és \$ i el preanàlisi és \$.

A aquest tipus d'analitzador els direm analitzadors sintàctics descendents predictius o analitzadors LL(1), i a la subclasse de gramàtiques independents del context que s'hi poden analitzar la direm classe LL(1).

Si augmentem el nombre de símbols de preanàlisi, parlarem de gramàtiques LL(2), LL(3), ...

Produccions ϵ

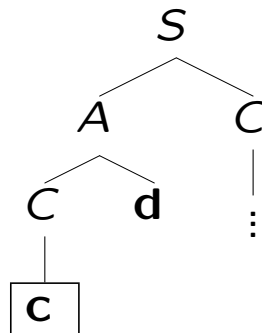
Tot i això, calcular els conjunts de predicció no és tan senzill en general com en els exemples anteriors.

G_4 :

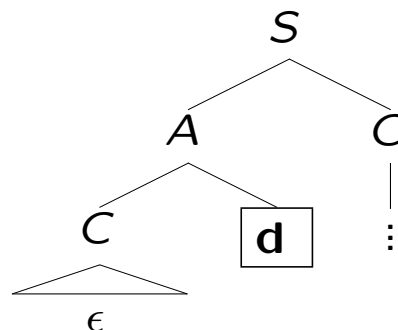
$$\begin{array}{lcl} S & \longrightarrow & A C \\ A & \longrightarrow & C d \mid \mathbf{b a} \\ C & \longrightarrow & \mathbf{c} \mid \epsilon \end{array}$$

Els subarbres de $A \rightarrow C d$ poden ser tres:

Subarbre 1:



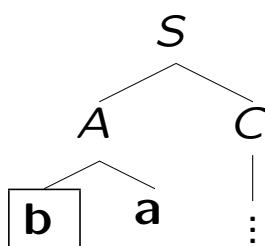
Subarbre 2:



Doncs, $\text{PRED}(A \rightarrow C d) = \{ \mathbf{c}, \mathbf{d} \}$.

Produccions ϵ

Subarbre 3:



I, per tant, $\text{PRED}(A \rightarrow \mathbf{b} \mathbf{a}) = \{ \mathbf{b} \}$.

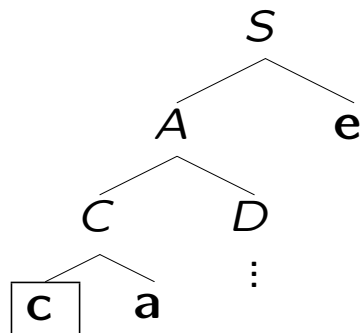
Subarbres buits

G_5 :

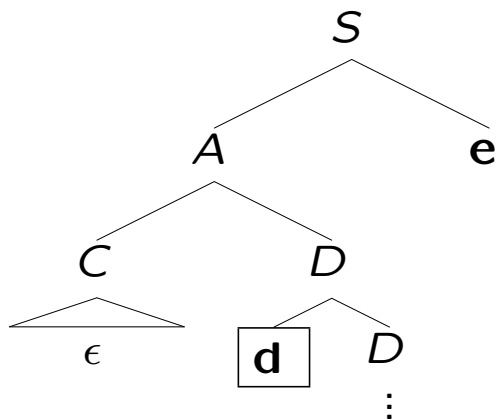
$$\begin{array}{lcl}
 S & \longrightarrow & A e \\
 A & \longrightarrow & C D \mid E c \\
 C & \longrightarrow & c a \mid \epsilon \\
 D & \longrightarrow & d D \mid \epsilon \\
 E & \longrightarrow & b
 \end{array}$$

Possibles arbres aplicant $A \rightarrow C D$:

Arbre 1:

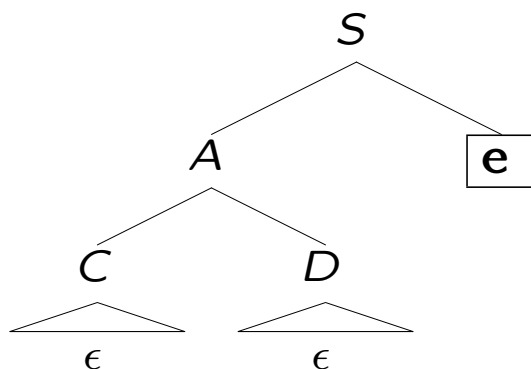


Arbre 2:



Subarbres buits

Arbre 3:



En aquest cas, doncs, $\text{PRED}(A \rightarrow C D) = \{ \mathbf{c}, \mathbf{d}, \mathbf{e} \}$.

Hem d'anar amb compte: el primer símbol no està sempre sota C (arbre 2); fins i tot, en l'últim cas (arbre 3) no està sota A .

Abans hem estudiat el PRIMER terminal que podia aparèixer sota A , però ara cal estudiar també el SEGÜENT terminal immediatament a la dreta de A .

Exercici: què passaria si les regles del no terminal E foren aquestes?

$$E \longrightarrow \mathbf{b} \mid \mathbf{e}$$

Recursivitat per l'esquerra

G_6 :

$$\begin{aligned} S &\longrightarrow AB \\ A &\longrightarrow Ac \mid a \\ B &\longrightarrow b \end{aligned}$$

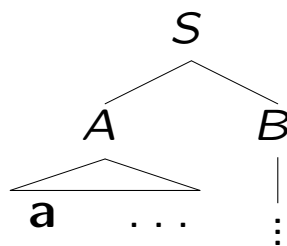
Cadenes del llenguatge:

a b \$, **a c b \$**, **a c c b \$**, ..., **a cⁿ b \$**, ...

Traça:

Pila	Entrada	Acció
\$ S	a c ⁿ b \$	expandir $S \rightarrow AB$
\$ B A	a c ⁿ b \$?

Arbre:



El cas base de la recursivitat (**a**) pertany als conjunts de predicció de les dues regles de A ahora! La gramàtica no és $LL(k)$ per a cap k finit...

Factors comuns per l'esquerra

G_7 :

$$\begin{array}{l} S \longrightarrow AB \\ A \longrightarrow aBc \mid aBe \\ B \longrightarrow b \end{array}$$

Ens trobem un problema quan el cim de la pila és A i el preanàlisi és a . G_7 és una gramàtica LL(3), però no LL(1).

Solució

- Si una gramàtica és
 - ambigua
 - o recursiva per l'esquerra
 - o té factors comuns per l'esquerra,no és LL(1).
- Cal trobar-ne una de nova que no ho siga i que genere el mateix llenguatge.

Implementació recursiva

- Un analitzador sintàctic descendent recursiu és una implementació que no fa servir cap pila explícita. Es basa en la pila de cridades a subprogrames (pila d'execució).
- Cada no terminal de la gramàtica té associat un subprograma que busca el símbol de preanàlisi en cada conjunt de predicció.

Implementació recursiva

G_8 :

$$\begin{array}{lcl} S & \longrightarrow & \mathbf{a} A D \\ A & \longrightarrow & D \mathbf{b} \mathbf{c} \mid \mathbf{e} \\ D & \longrightarrow & \mathbf{d} \end{array}$$

global preanàlisi

funció emparella (token)

si token = preanàlisi
preanàlisi := analex()

si no
error: trobat preanàlisi, esperava token

funció $S()$

si preanàlisi \in PRED($S \rightarrow \mathbf{a}AD$)
emparella(**a**); $A()$; $D()$

si no
error: trobat preanàlisi, esperava PRED($S \rightarrow \mathbf{a}AD$)

funció $A()$

si preanàlisi \in PRED($A \rightarrow D\mathbf{b}c$)
 $D()$; emparella(**b**); emparella(**c**)

si no
si preanàlisi \in PRED($A \rightarrow \mathbf{e}$)
emparella(**e**)

si no
error: trobat preanàlisi,
esperava PRED($A \rightarrow D\mathbf{b}c$) \cup PRED($A \rightarrow \mathbf{e}$)

funció $D()$...

principal

preanàlisi := analex()
 $S()$

si preanàlisi \neq \$
error: trobat preanàlisi,
esperava fin d'entrada

A continuació...

Com podem calcular els conjunts de predicció de forma algorísmica?